

## パズル「ナンバープレイス」の難易度について(2)

著者	佐藤 金吾
出版者	法政大学多摩研究報告編集委員会
雑誌名	法政大学多摩研究報告
巻	17
ページ	13-42
発行年	2002-03-30
URL	<a href="http://hdl.handle.net/10114/1593">http://hdl.handle.net/10114/1593</a>

# パズル「ナンバープレース」の難易度について (2)

佐藤金吾

On characteristic of difficult degree to solve Number-Place puzzles(2)

Kingo SATO

## 1. はじめに

小論[1]において「ナンバープレース」の難易度の基準についてある結果を出した。しかし、5段階レベルの難易基準においては、上位3レベルを分ける明確な基準が示せなかった。それに  
つき、7節の今後の課題として、問題の本来持っている難しさややっかいさに加えて、解くとき  
の手間とか面倒くささを考慮すべきことをあげた。

以下の考察ではこの点が核になっている。従って、5段階レベルの難易に対する基準が主な  
る対象であるが、5節で難易度6と7の問題についても若干考察する。

なお、これからの議論で必要なことを2つほど記しておく。

① 実際の問題についての解の結果が必要なので、問題例として、[1]でも使用した[2],[3]の  
178題を用いた。さらに、5節では新しいもの[4]を用いた。

② 本文中のプログラムには、言語として「構造化 BASIC98」を用いた。

## 2. 難易度の基準 (1)

### 2.1 レベル1とレベル2の基準

178題の中には、レベル1が58題、レベル2が58題、そしてレベル3が50題ある。

難易度の核になる「解決に必要な空きマス技法」についてまとめるとつぎとなる。

	1-空きマス 技法まで	2-空きマス 技法まで	3-空きマス 技法まで	4-空きマス 技法まで	5-空きマス 技法まで	6-空きマス 技法以上	計
レベル1	36	20	0	2	0	0	58
レベル2	6	21	22	6	3	0	58
レベル3	0	4	22	18	2	4	50

この結果から、つぎのことがいえる。

「解決に必要な  $n$  - 空きマス技法の  $n$  の値により、

$1 \leq n \leq 2$  ならレベル 1、 $2 \leq n \leq 3$  ならレベル 2、 $3 \leq n \leq 4$  ならレベル 3 とする。」

実際、この基準をみたすものの割合はレベル 1 で 97 %、レベル 2 で 74 %、レベル 3 で 80 % である。

しかし、 $n$  の値でのダブリがあつて一意にレベルが決められないので、これに

「解決までに調べた  $n$  - 空きマス技法の回数」( $n$  - チェック数と呼ぶ)を加える。このチェック数は「解くときの手間」を考慮したものである。

1) 2-空きマス技法まで必要な問題についての 2-チェック数の度数表

チェック数	1回	2回	3回	4回	5回	6回	7回以上	計
レベル 1	8	4	3	3	0	0	2	20
レベル 2	5	6	3	2	1	1	3	21

2) 3-空きマス技法まで必要な問題についての 3-チェック数の度数表

チェック数	1回	2回	3回	4回	5回	6回	7回以上	計
レベル 2	6	5	4	2	2	3	0	22
レベル 3	2	3	3	3	0	7	4	22

以上の結果を踏まえ、つぎの基準が得られる。

[基準 1] 解決に必要な  $n$  - 空きマス技法の  $n$  の値とその  $n$  - チェック数の回数により、

1)  $n = 1$  であるか、あるいは  $n = 2$  かつ 2-チェック数が 2 回以下ならレベル 1、

2)  $n = 2$  かつ 2-チェック数が 3 回以上であるか、あるいは  $n = 3$  かつ 3-チェック数が 3 回以下ならレベル 2、

3)  $n = 3$  かつ 3-チェック数が 4 回以上ならレベル 3 とする。

(注意 2.1)

1)  $n = 3$  に関しこのチェック数に関する基準をみたす割合は、レベル 2 で 68 %、レベル 3 で 64 % である。

2) しかし、 $n = 2$  に関しこのチェック数に関する基準をみたす割合は、レベル 1 で 60 %、レベル 2 で 48 % と、かなり無理がある。これはあえてダブらない線引きをした結果である。上で得られた解答結果をよく反映する基準としては、ダブってしまうが、「2-チェック数が 4 回以下ならレベル 1、2-チェック数が 2 回以上ならレベル 2」が考えられる。この基準をみたす割合はレベル 1 で 90 %、レベル 2 で 76 % となる。

## 2.2 上位3レベルの基準

上位3レベル、すなわちレベル3, 4, そして5を分ける基準を考えるために、[1]の6節で得られた結果をここにあげる。

	3-空きマス 技法まで	4-空きマス 技法まで	5-空きマス 技法まで	技法4 が必要	技法5 が必要	技法6 が必要	技法7 使用	計
レベル3	26	18	3	0	0	0	3	50
レベル4	7	18	11	4	2	0	10	52
レベル5	0	12	9	9	0	1	15	46
計	33	48	23	13	2	1	28	148

この結果から、つぎのことがいえる。

「解決に必要な  $n$  - 空きマス技法の  $n$  の値により、

$3 \leq n \leq 4$  ならレベル3、 $4 \leq n$  ならレベル4、 $5 \leq n$  ならレベル5とする。」

しかし、 $n$  の値でのダブリがあって一意にレベルが決められないので、2節と同様にチェック数を考慮する。また、つぎの用語を定義する。

(定義) 4 - 空きマス技法まででは解けないものを、**hard** - タイプとよぶ。

1) 4-空きマス技法まで必要な問題についての4-チェック数の度数表

チェック数	1回	2回	3回	4回	5回	6回	7回	8回	9回	10回以上	計
レベル3	2	3	5	0	2	1	0	2	2	1	18
レベル4	2	1	2	2	3	2	1	2	0	3	18

以上の結果および2節の結果を踏まえ、つぎの基準が得られる。

[基準II] 解決に必要な  $n$  - 空きマス技法の  $n$  の値とその  $n$  - チェック数の回数により、

1)  $n = 3$  かつ3 - チェック数が4回以上であるか、あるいは  $n = 4$  かつ4 - チェック数が3回以下ならレベル3、

2)  $n = 4$  かつ4 - チェック数が4回以上であるか、あるいは **hard** - タイプならレベル4、

3) **hard** - タイプならレベル5とする。

(注意2.2)

1)  $n = 4$  に関しこのチェック数に関する基準をみたす割合は、レベル3で56%、レベル4で72%である。

2) **hard** - タイプについて、レベル4とレベル5の区分ができていない。この条件を求めるのが3節以降の課題となる。

### 3. 新しい解き方と基準化のための評価値

#### 3.1 新しい解き方の導入

2節の結果から、5段階レベルの難易度に対する完全な基準を得るには *hard* - タイプについてレベル4と5を分ける明確な基準が必要である。このためには、はじめの所で述べたように新しい取組み方が必要であり、[1]で示した解き方を、

- ① 人間のやり方により近い方法に修正し、
- ② 解くときの手間とか面倒くささを加えなければならない。

さて、以下ではつぎの解き方を採用する。

[新しい解き方]

1) 技法3として、4 - 空きマス技法までを用いる。従って、5以上の空きマス技法は使わない。

2) 技法4～7の内、技法4と技法7のみ使用する。

ただし、技法4の中で用いるのは3 - 空きマスまでに限る。

3) 技法3と技法4のチェック数および技法7の使用回数を基準の評価に加える。

ただし、技法3では3 - チェック数と4 - チェック数、技法4では2 - チェック数と3 - チェック数だけを用いる。

上記の決め方の理由は、1)についてはこの方が人間のやり方により近いと考えられるからであり、2)については148題の中で技法5, 6が使われたのはわずか3題だから、さらに人間のやり方を考えると技法4での4 - 空きマス使用は少々無理と思われるからであり、3)についてはすでに2節や3節でみたように、解くときの手間とか面倒くささをこれらの回数が表すからである。

(注意3.1)

1) 技法3として、3 - 空きマス技法で止めないで4 - 空きマス技法までを用いる理由は、148題のうち48題(約32%)が4 - 空きマス技法で解けるからである。

2) この新しい解き方で中心になるのは技法7、すなわち、

「2つの数字(数字対)が入る可能性をもつ空きマスを選び、その一方の数字を仮に入れて解を求める処理を続ける。このとき、

- ① 完成すれば、それが求める解である。
- ② 途中で矛盾が出れば、仮の数字の対である他方の数字がこのマスの表示数として決まる。
- ③ 矛盾が出ないまま継続作業がストップしたら、対で可能な新しい空きマスに代えて、上記処理を新規に行なう。」

### 3.2 対リストの作成

[1]で用いた技法7のやり方は人間のする仕方とはだいぶ異なるので、新しい解き方にふさわしい方法を導入する。ここで用いるのは対リストを作成する方法である。すなわち、数字対と場所対のリストを作っておき、そこから順番に候補者を取り出し試していくのだが、見つけ易い数字対や場所対が前に、見つけ難いものが後にくるようにすれば、同時に難易評価もできることになる。

さて、対を見つける3通りの方法を次にあげるが、後のものほど見つけ難くなっている。

- 1) 各タテ列、各ヨコ列そして各ブロックの中で空きマスが2個のもの
- 2) 各グループ確定数による補－空きマス群の中で空きマスが2個のもの
- 3) 技法2において2ペア数の置ける場所が2つのもの

なお、2), 3)から得られた対を $\beta$ -対とよぶ。

### 3.3 基準化のための評価値

新しい解き方に応じた評価値を決めるにあたり、基本的に[1]の6節で求めたものを若干手直しして用いることにする。すなわち、

- 1) 1回あたりの使用（手間）ポイントをつぎとする。
  - ① 3-空きマス技法：11.5      ② 4-空きマス技法：17
  - ③ 技法4での    2-空きマス技法：8    3-空きマス技法：15.5
  - ④ 技法7の使用：4

2) 評価値は解くまでに使用した上記技法のポイントをすべて加え、さらにわかり易いように10で割って四捨五入して求める。

(注意3.2) 1) 上で2-空きマス技法のポイントを考慮しなかったのは、対象になるタテ列、ヨコ列、ブロック列が複数あるとき、その優先順位の決め方によってその使用回数が一番変化しやすいこと、及びその値を加えても評価値全体にほとんど影響を与えないことが見られるので、除外したからである。

2) 新しい解き方及び評価値の計算のコンピュータプログラムは、レベルを計算するプログラムとして次節で示す。

## 4. 難易度の基準 (2)

4.1 レベル4とレベル5への分離基準が必要な *hard* - タイプの61題の結果はつぎのようにまとめられる。

- 1) 問題の持つ難しさについて

解くのに技法7を多く必要とするのは難しさの一面であるといえる。

技法7の使用回数	なし	1回	2回	3回以上	計
レベル4	6	15	5	1	27
レベル5	7	15	7	5	34

この結果は、上記のことを示しているとみられる。また、 $\beta$ -対を使わないと解けないというのも難しさの一面であろう。これについては、

レベル4(技法7使用は21題)では4題、すなわちその約19%、

レベル5(技法7使用は27題)では9題、すなわちその約33%

がこの条件をみたし、やはり難しさの一面を示している。

## 2) 問題を解くのにかかる手間や面倒くささ

3節で考察した評価値がこの尺度を示すが、61題の評価値の度数分布表はつぎとなる。ただし、人間の仕方に近づけた新しい解き方のプログラムでは解けない問題が1題あり、これは便宜上、最後のクラスに入れた。

評価値	10～	20～	30～	40～	50～	100～	計
レベル4	1	5	8	5	6	2	27
レベル5	2	6	5	2	13	6	34

以上の結果を踏まえ、つぎの基準が得られる。

[基準Ⅲ] *hard* - タイプの問題に対し、

1) 解けないもの、及び技法7で $\beta$ -対を用いるものをレベル5とする。

2) 評価値が50未満のものをレベル4、50以上のものをレベル5とする。

(注意4.1) 上の基準をみたすものは、レベル4で63%、レベル5で59%である。

## 4.2 レベルを計算するコンピュータプログラム

```

1000 REM *** Number Place ***
1010 REM ===== nanido_kijyun =====
1020 DEF FNBL(I)=INT((I-1)/3)+1
1030 CONSOLE .,0,1: SCREEN 3: CLS 3
1040 ' --- hairetu_sengen ---
1050 DIM M(9,9),GM(9,9),HM(9,9),A(9),AA(9,9),GS(9),N(9)
1060 DIM GTY(60),GI(60),GJ(60),GSU(60),AP(3),SG(4),P(9),PI(9,5),PJ(9,5)
1070 DIM GRTY(60),GRI(60),GRJ(60),GRD(60),GRP(60),GRK(60),GRIT(60,3),GRSU(60,3)
1080 DIM ITI(2),TSU(9),KOSU(2)
1090 DIM MSU(2,6),NT(6,27),SYON(6)
1100 DIM PAIRI(45,2),PAIRS(45,2),PRI(2),PRS(2),HYOT(8),KIJYUN$(7)

```

```
1110 ' --- syoki_settei ---
1120 GKAKUTEI=0
1130 FOR I=1 TO 7: READ KIYUN$(I): NEXT I
1140 DATA レベル 1,レベル 2,レベル 3,レベル 4,レベル 5,レベル 6,レベル 7
1150 ' --- data_yomi & syoki_jyoho-input ---
1170 GOSUB *DATAYOMI
1180 '
1190 FOR TA=1 TO 9:FOR YO=1 TO 9
1200 IF GM(TA,YO)>0 THEN GKAKUTEI=GKAKUTEI+1
1210 NEXT YO,TA
1220 ' --- su_kakutei ---
1230 FKAI=0: SUPHARD=0:MGIHO=1
1240 WHILE FKAI=0
1250 ' -- shoki_settei --
1260 KAKUTEI=GKAKUTEI
1270 FOR I=1 TO 9:FOR J=1 TO 9: M(I,J)=GM(I,J): NEXT J,I
1280 FOR I=1 TO 2:FOR J=1 TO 6: MSU(I,J)=0: NEXT J,I
1290 FOR I=1 TO 6: SYON(I)=0: NEXT I
1300 '
1310 FGIHOU7=0:FMUJYUN=0:MAKIN=0:MGH7TY=0:FKANO=0:MGH7SU=0
1320 IF SUPHARD=0 THEN AKISU=4:FGH7C=1 ELSE AKISU=5:FGH7C=2
1330 '
1340 FSEACH=0
1350 WHILE FSEACH=0
1360 GOSUB *NTSET
1370 '
1380 FTYPE1END=0
1390 WHILE FTYPE1END=0
1400 ' -- gihou_2,3 --
1410 FGIHO23END=0:GIHOTY=1
1420 WHILE FGIHO23END=0
1430 TUIKA=0
1440 FOR NAKI=1 TO AKISU
1450 IF NAKI=1 THEN
1460 GOSUB *GIHOU3
1470 IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP1
1480 FH1=0:GOSUB *GIHOU2
```



```
1490     IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP1
1500     ELSE
1510         GOSUB *GIHOU3
1520         IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP1
1530     END IF
1540     NEXT NAKI
1550     *OUTLOOP1
1560     IF (KAKUTEI=81)OR(TUIKA=0)OR(FMUJYUN=1) THEN FGIHO23END=1
1570     WEND
1580     IF KAKUTEI=81 THEN GOTO *KAIEND
1590     IF FMUJYUN=1 THEN GOTO *GIHOU7
1600     ' -- gihou_4 --
1610     ' group_kaku sakusei
1620     GOSUB *MAKEGROUP
1630     GIHOTY=2
1640     FOR NAKI=1 TO AKISU-1
1650         GOSUB *GIHOU4
1660         IF (TUIKA=1)AND(MGIHO=1) THEN MGIHO=2
1670         IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP2
1680     NEXT NAKI
1690     *OUTLOOP2
1700     IF (KAKUTEI=81)OR(TUIKA=0)OR(FMUJYUN=1) THEN FTYPE1END=1
1710     WEND
1720     IF KAKUTEI=81 THEN GOTO *KAIEND
1730     *GIHOU7
1740     ' -- gihou_7 --
1750     IF FGH7C=1 THEN
1755         ' -- gihou_7A --
1760         MGIHO=3
1770         IF FGIHOU7=0 THEN SENUM=0: GOSUB *PAIRLIST
1780         '
1790         IF FGIHOU7=0 THEN
1800             FGIHOU7=1:FMUJYUN=0
1810             ' hikaе_toru
1820             HKAKUTEI=KAKUTEI
1830             FOR I=1 TO 9:FOR J=1 TO 9: HM(I,J)=M(I,J): NEXT J,I
1840             ' koho_set
```

```
1850  GOSUB *KOHOSSET
1860  ELSE
1870    ' atai_modosi
1880  KAKUTEI=HKAKUTEI
1890  FOR I=1 TO 9:FOR J=1 TO 9: M(I,J)=HM(I,J): NEXT J,I
1900    ' mujyun_ari?
1910  IF FMUJYUN=0 THEN
1920    GOSUB *KOHOSSET
1930  ELSE
1940    FGIHOU7=0:FMUJYUN=0
1950    ' kakutei_syori
1960    TA=INT((PAIRI(SENUM,2)-1)/9)+1:YO=((PAIRI(SENUM,2)-1) MOD 9)+1
1970    SU=PAIRS(SENUM,1):GOSUB *KAKUTEI
1980  END IF
1990  END IF
2000  IF FKANO=1 THEN FSEACH=1: SUPHARD=1:MGIHO=4
2010  ELSE
2020    ' -- gihou_7B --
2030  IF FGIHOU7=0 THEN
2040    FGIHOU7=1:FMUJYUN=0
2050    ' hikaet_toru
2060  HKAKUTEI=KAKUTEI
2070  FOR I=1 TO 9:FOR J=1 TO 9: HM(I,J)=M(I,J): NEXT J,I
2080    ' koho_sentak
2090  KOTA=1:KOYO=0: GOSUB *KOHOSSEN
2100  ELSE
2110    ' atai_modosi
2120  KAKUTEI=HKAKUTEI
2130  FOR I=1 TO 9:FOR J=1 TO 9: M(I,J)=HM(I,J): NEXT J,I
2140    ' mujyun_ari?
2150  IF FMUJYUN=0 THEN
2160    GOSUB *KOHOSSEN
2170  ELSE
2180    FGIHOU7=0:FMUJYUN=0
2190    ' kakutei_syori
2200    TA=KOTA:YO=KOYO:SU=KOSU(2): GOSUB *KAKUTEI
2210  END IF
```

```
2220 END IF
2230 IF FKANO=1 THEN FSEACH=1:FKAI=1
2240 END IF
2250 WEND
2260 WEND
2270 *KAIEND
2280 ' --- kai_hyoji ---
2290 GOSUB *KAIHYOJI
2300 GOSUB *KIJYUN
2310 END
2320 :::::::
3000 *DATAYOMI
3010 FILE$=" a:npprob.dat"
3020 OPEN FILE$ FOR INPUT AS #1
3030 '
3040 FOR I=1 TO 9:FOR J=1 TO 9
3050 GM(I,J)=0
3060 NEXT J,I
3070 WHILE NOT EOF(1)
3080 INPUT #1,TA,YO,SU
3090 GM(TA,YO)=SU
3100 WEND
3110 RETURN
3120 ::
3200 *NTSET
3210 FOR I=1 TO 6
3220 N9=0
3230 FOR J=1 TO 3:FOR K=1 TO 9
3240 N9=N9+1: NT(I,N9)=10*J+K
3250 NEXT K,J
3260 NEXT I
3270 RETURN
3280 ::
3300 *KAIHYOJI
3310 XGS=168:XGL=384
3320 YGS=40:YGL=328
3330 ' -- waku_hyouji --
```

```
3340 FOR YL=YGS TO YGL STEP 32
3350 LINE (XGS,YL)-(XGL,YL)
3360 IF (YL-YGS) MOD 96=0 THEN LINE (XGS,YL+1)-(XGL,YL+1)
3370 NEXT YL
3380 FOR XL=XGS TO XGL STEP 24
3390 LINE (XL,YGS)-(XL,YGL)
3400 IF (XL-XGS) MOD 72=0 THEN LINE (XL+1,YGS)-(XL+1,YGL)
3410 NEXT XL
3420 ' -- suu_hyouji --
3430 FOR TA=1 TO 9:FOR YO=1 TO 9
3440 IF M(TA,YO)>0 THEN
3450 XHYJ=21+3*(YO-1): YHYJ=3+2*(TA-1)
3460 LOCATE XHYJ,YHYJ: PRINT M(TA,YO);
3470 END IF
3480 NEXT YO,TA
3490 RETURN
3495 :::
3500 *KAKUTEI
3510 ' == par. TA,YO,SU,BUNRN ==
3520 KAKUTEI=KAKUTEI+1: TUIKA=1
3530 M(TA,YO)=SU
3540 RETURN
3550 :::
3600 *HOSIKI2
3610 ' == par. TA,YO ==
3620 GOSUB *SIYOUSUU
3630 IF PN=8 THEN SU=KN:GOSUB *KAKUTEI
3640 RETURN
3650 :::
3660 *SIYOUSUU
3670 FOR U=1 TO 9: N(U)=0: NEXT U
3680 '
3690 FOR U=1 TO 9: N(M(TA,U))=1: NEXT U
3700 FOR U=1 TO 9: N(M(U,YO))=1: NEXT U
3710 B1=FNBL(TA):B2=FNBL(YO)
3720 FOR U1=3*B1-2 TO 3*B1:FOR U2=3*B2-2 TO 3*B2
3730 N(M(U1,U2))=1
```

```
3740 NEXT U2,U1
3750 '
3760 PN=0:SYS=0
3770 FOR U=1 TO 9
3780 IF N(U)=1 THEN PN=PN+1 ELSE KN=U:SYS=SYS+1:TSU(SYS)=U
3790 NEXT U
3800 RETURN
3810 :::
3900 *BLOCKSU
3910 REP=1
3920 WHILE REP=1
3930 REP=0: H4=0
3940 FH1=1:GOSUB *GIHOU2
3950 IF TUIKA=1 THEN REP=1
3960 WEND
3970 RETURN
3980 :::::
4000 *GIHOU2
4010 ' == par. FH1 ==
4020 GIHOU=2
4030 FOR TY=1 TO 2
4040 ' ty=1 -- yoko ty=2 -- tate
4050 FOR II=1 TO 3
4060 TE2TY=1:GOSUB *TECH2
4070 IF (FH1=0)AND(TUIKA=1) THEN RETURN
4080 NEXT II
4090 NEXT TY
4100 RETURN
4110 ::
4220 *TECH2
4230 ' == par. TE2TY,FH1,TY,II ==
4240 ' syoki_ka
4250 FOR I9=1 TO 9
4260 P(I9)=0
4270 FOR J9=0 TO 3: PI(I9,J9)=0:PJ(I9,J9)=0: NEXT J9
4280 NEXT I9
4290 '

```

```
4300  FOR I9=3*II-2 TO 3*II
4310  FOR J9=1 TO 9
4320    IF TY=1 THEN MS=M(I9,J9) ELSE MS=M(J9,I9)
4330    IF MS>0 THEN
4340      P(MS)=P(MS)+1
4350      PI(MS,P(MS))=I9:PJ(MS,P(MS))=FNBL(J9)
4360    END IF
4370  NEXT J9
4380 NEXT I9
4390 '
4400 FOR SU=1 TO 9
4410 IF P(SU)=2 THEN
4420   GRI=9*(II-1)+6-(PI(SU,1)+PI(SU,2)):GRJ=6-(PJ(SU,1)+PJ(SU,2))
4430   PN=0: PNN=0
4440   FOR J9=3*GRJ-2 TO 3*GRJ
4450     IF TY=1 THEN TA=GRI:YO=J9 ELSE TA=J9:YO=GRI
4460     IF M(TA,YO)>0 THEN
4470       PN=PN+1
4480     ELSE
4490       FOR J8=1 TO 9
4500         IF TY=1 THEN MS=M(J8,YO) ELSE MS=M(TA,J8)
4510         IF MS=SU THEN PN=PN+1:GOTO *EXJ8
4520       NEXT J8
4530       JN=J9
4540       PNN=PNN+1: A(PNN)=J9
4550       *EXJ8
4560     END IF
4570   NEXT J9
4580 '
4590 SELECT TE2TY
4600 CASE 1
4610   IF PN=2 THEN
4620     IF TY=1 THEN TA=GRI:YO=JN ELSE TA=JN:YO=GRI
4630     IF FH1=0 THEN GOSUB *KAKUTEI: RETURN
4640   ELSE
4650     IF FH1=1 THEN
4660       H4=H4+1
```

```
4670      GTY(H4)=TY: GSU(H4)=SU: GI(H4)=GRI:GJ(H4)=GRJ
4680      END IF
4690      END IF
4700      CASE 2
4710      IF PN=1 THEN
4720      FOR I5=1 TO 2
4730      IF TY=1 THEN TA=GRI:YO=A(I5) ELSE TA=A(I5):YO=GRI
4740      PRI(I5)=9*(TA-1)+YO: PRS(I5)=SU
4750      NEXT I5
4760      ' kiroku
4770      GOSUB *KIROKU
4780      END IF
4790      END SELECT
4800      END IF
4810      NEXT SU
4820      RETURN
4830      ::::
5000 *GIHOU3
5010      GIHOU=3
5020      IF NAKI>MAKIN THEN MAKIN=NAKI
5030      FOR G9=0 TO 26
5040      PG9=((G9+SYON(NAKI)) MOD 27)+1
5050      TTY=INT(NT(NAKI,PG9)/10): I=NT(NAKI,PG9) MOD 10
5060      IF TTY=3 THEN BI=INT((I-1)/3)+1: BJ=((I-1) MOD 3)+1
5070      GOSUB *TECH3
5080      IF TUIKA=1 THEN GOSUB *SYOSET: RETURN
5090      NEXT G9
5100      RETURN
5110      :::
5120      *SYOSET
5130      SYON(NAKI)=PG9
5140      IF (NAKI=4)AND(PG9-1<SYON(3)) THEN
5150      SWAP NT(3,PG9),NT(3,SYON(3)): SYON(3)=SYON(3)-1
5160      END IF
5170      RETURN
5180      :::
5200 *TECH3
```

```
5210 ' == par. TTY,I(or BI,BJ) ==
5220 ' aki_check
5230 GOSUB *AKICH
5240 ' -- kakutei_syori --
5250 IF AKI=NAKI THEN
5260 ' kiroku
5270 MSU(GIHOTY,AKI)=MSU(GIHOTY,AKI)+1
5280 ' -- hosiki_1 --
5290 FOR S9=1 TO AKI
5300 SU=GS(S9)
5310 PN=0
5320 SELECT TTY
5330 CASE 1,2
5340 FOR J9=1 TO AKI
5350 IF TTY=1 THEN TA=I:YO=A(J9) ELSE TA=A(J9):YO=I
5360 GOSUB *ITTICH
5370 IF FITTI=1 THEN PN=PN+1 ELSE JN=A(J9)
5380 NEXT J9
5390 IF TTY=1 THEN TA=I:YO=JN ELSE TA=JN:YO=I
5400 CASE 3
5410 FOR K9=1 TO AKI
5420 TA=AA(K9,1):YO=AA(K9,2): GOSUB *ITTICH
5430 IF FITTI=1 THEN PN=PN+1 ELSE IN=AA(K9,1):JN=AA(K9,2)
5440 NEXT K9
5450 TA=IN:YO=JN
5460 END SELECT
5470 '
5480 IF PN=AKI-1 THEN
5490 GOSUB *KAKUTEI
5500 RETURN
5510 END IF
5520 ' -- mujyun_check --
5530 IF (FGIHOU7=1)AND(PN=AKI) THEN FMUJYUN=1: RETURN
5540 NEXT S9
5550 ' -- hosiki_2 --
5560 FOR K9=1 TO AKI
5570 IF TTY=1 THEN TA=I:YO=A(K9)
```



```
5580   IF TTY=2 THEN TA=A(K9):YO=I
5590   IF TTY=3 THEN TA=AA(K9,1):YO=AA(K9,2)
5600   GOSUB *HOSIKI2
5610   IF TUIKA=1 THEN RETURN
5620   NEXT K9
5630 END IF
5640 RETURN
5650 ::::
5660 *AKICH
5670 ' == par. TTY,I ==
5680 FOR I9=1 TO 9: N(I9)=0: NEXT I9
5690 '
5700 AKI=0
5710 SELECT TTY
5720   CASE 1,2
5730     FOR J9=1 TO 9
5740       IF TTY=1 THEN MS=M(I,J9) ELSE MS=M(J9,I)
5750       IF MS=0 THEN AKI=AKI+1:A(AKI)=J9 ELSE N(MS)=1
5760     NEXT J9
5770   CASE 3
5780     FOR I9=3*BI-2 TO 3*BI:FOR J9=3*BJ-2 TO 3*BJ
5790       IF M(I9,J9)=0 THEN AKI=AKI+1:AA(AKI,1)=I9:AA(AKI,2)=J9 ELSE
N(M(I9,J9))=1
5800     NEXT J9,I9
5810   END SELECT
5820 '
5830 GSN=0
5840 FOR I9=1 TO 9
5850   IF N(I9)=0 THEN GSN=GSN+1:GS(GSN)=I9
5860 NEXT I9
5870 RETURN
5880 :::
5890 *ITTICH
5900 ' == par. TA,YO,SU ==
5910 FITTI=0
5920 FOR U9=1 TO 9
5930   IF M(TA,U9)=SU THEN FITTI=1
```

```
5940   IF M(U9,YO)=SU THEN FITTI=1
5950   NEXT U9
5960   B1=INT((TA-1)/3)+1:B2=INT((YO-1)/3)+1
5970   FOR U9=3*B1-2 TO 3*B1:FOR U8=3*B2-2 TO 3*B2
5980     IF M(U9,U8)=SU THEN FITTI=1
5990   NEXT U8,U9
6000   RETURN
6010   :::
6020 *BLNARCH
6030   GSY=0
6040   IF TTY=3 THEN
6050     FOR GR=1 TO 2
6060       BEN=0: HB=AA(1,GR)
6070       FOR K9=2 TO AKI
6080         IF HB<>AA(K9,GR) THEN BEN=1
6090       NEXT K9
6100       IF BEN=0 THEN GSY=1:HGR=GR
6110     NEXT GR
6120   ELSE
6130     BEN=0: HB=INT((A(1)-1)/3)+1
6140     FOR K9=2 TO AKI
6150       IF HB<>INT((A(K9)-1)/3)+1 THEN BEN=1
6160     NEXT K9
6170     IF BEN=0 THEN GSY=1
6180   END IF
6190   RETURN
6195   :::
6200 *MAKEGROUP
6210   GKO=0
6220   GOSUB *LISTGRP1
6230   FH1=1:GOSUB *BLOCKSU
6240   IF H4>0 THEN GOSUB *LISTGRP2
6250   ' ho_group sakusei
6260   HOD=0:GOSUB *HOGROUP
6270   HOD=1:GOSUB *HOGROUP
6280   GOSUB *DBHOGRP
6290   RETURN
```

```
6295 :::
6300 *LISTGRP1
6310 MKGRPN=1
6320 ' tate
6330 FOR I=1 TO 9
6340   TTY=2:GOSUB *MKGRP
6350 NEXT I
6360 ' yoko
6370 FOR I=1 TO 9
6380   TTY=1:GOSUB *MKGRP
6390 NEXT I
6400 ' block
6410 FOR BI=1 TO 3:FOR BJ=1 TO 3
6420   TTY=3: GOSUB *MKGRP
6430 NEXT BJ,BI
6440 RETURN
6450 :::
6500 *LISTGRP2
6510 FOR W=1 TO H4-1
6520   TTY=GTY(W):GI=GI(W):GJ=GJ(W):SU=GSU(W)
6530   ' kanou_su
6540   CN=1: SG(1)=SU
6550   FOR J9=W+1 TO H4
6560     IF (TTY=GTY(J9))AND(GI=GI(J9))AND(GJ=GJ(J9)) THEN
6570       CN=CN+1:SG(CN)=GSU(J9)
6580     END IF
6590   NEXT J9
6600   '
6610   IF CN>1 THEN
6620     ' jyogai_su
6630     JGSU=0
6640     FOR K9=1 TO 3
6650       GJK=3*(GJ-1)+K9
6660       IF TTY=1 THEN MS=M(GI,GJK) ELSE MS=M(GJK,GI)
6670       IF MS>0 THEN
6680         JGSU=JGSU+1:JG(JGSU)=K9
6690       ELSE
```

```
6700      CIT=0
6710      FOR KP=1 TO CN
6720          FOR Q8=1 TO 9
6730              IF TTY=1 THEN MS=M(Q8,GJK) ELSE MS=M(GJK,Q8)
6740              IF MS=SG(KP) THEN CIT=CIT+1
6750          NEXT Q8
6760      NEXT KP
6770      IF CIT=CN THEN JGSU=JGSU+1:JG(JGSU)=K9
6780      END IF
6790  NEXT K9
6800  IF CN=3-JGSU THEN
6810      FOR K8=1 TO 3: N(K8)=1: NEXT K8
6820      FOR K8=1 TO JGSU: N(JG(K8))=0: NEXT K8
6830      APN=0
6840      FOR K8=1 TO 3
6850          IF N(K8)=1 THEN APN=APN+1:AP(APN)=3*(GJ-1)+K8
6860      NEXT K8
6870      ' kiroku
6880      GKO=GKO+1
6890      GRD(GKO)=0:GRP(GKO)=3
6900      GRTY(GKO)=TTY:GRI(GKO)=GI:GRJ(GKO)=GJ
6910      GRK(GKO)=CN
6920      FOR K9=1 TO CN
6930          GRIT(GKO,K9)=AP(K9):GRSU(GKO,K9)=SG(K9)
6940      NEXT K9
6950      END IF
6960  END IF
6970 NEXT W
6980 RETURN
6990 :::
7000 *HOGROUP
7010 MKGRP=2:SGKO=GKO
7020 FOR W=1 TO SGKO
7030     IF GRD(W)=HOD THEN
7040         TTY=GRTY(W):GI=GRI(W):GJ=GRJ(W):GD=GRD(W):GP=GRP(W)
7050         ' kari_oki
7060         FOR K9=1 TO GRK(W)
```

```

7070     IF TTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
7080     M(TA,YO)=GRSU(W,K9)
7090     NEXT K9
7100     KTY=TTY
7110     ' retu, block
7120     I=GI:GOSUB *MKGRP
7130     TTY=3:BI=INT((GI-1)/3)+1:BJ=GJ
7140     GOSUB *MKGRP
7150     ' M_modosi
7160     FOR K9=1 TO GRK(W)
7170         IF KTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
7180         M(TA,YO)=0
7190     NEXT K9
7200     END IF
7210 NEXT W
7220 RETURN
7230 :::::
7300 *DBHOGRP
7310 MKGRPN=3:SGKO=GKO
7320 FOR W=1 TO SGKO-1
7330     DTY=GRTY(W):DI=GRI(W):DJ=GRJ(W):DP=GRP(W)
7340     ' kari_oki
7350     FOR K9=1 TO GRK(W)
7360         IF DTY=1 THEN TA=DI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=DI
7370         M(TA,YO)=GRSU(W,K9)
7380     NEXT K9
7390     KDTY=DTY
7400     ' retu
7410     FOR X=W+1 TO SGKO
7420         IF (DTY=GRTY(X))AND(DI=GRI(X)) THEN
7430             TTY=GRTY(X):GI=GRI(X):GJ=GRJ(X):GP=GRP(X)
7440             ' kari_oki
7450             FOR K9=1 TO GRK(X)
7460                 IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9):YO=GI
7470                 M(TA,YO)=GRSU(X,K9)
7480             NEXT K9
7490             '

```

```
7500  I=GI:GOSUB *MKGRP
7510  ' M_modosi
7520  FOR K9=1 TO GRK(X)
7530    IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9):YO=GI
7540    M(TA,YO)=0
7550  NEXT K9
7560  END IF
7570  NEXT X
7580  ' block
7590  IF DTY=1 THEN DBI=FNBL(DI):DBJ=DJ ELSE DBI=DJ:DBJ=FNBL(DI)
7600  FOR X=W+1 TO SGKO
7610    TTY=GRTY(X):GI=GRI(X):GJ=GRJ(X)
7620    IF TTY=1 THEN BI=FNBL(GI):BJ=GJ ELSE BI=GJ:BJ=FNBL(GI)
7630    IF (DBI=BI)AND(DBJ=BJ) THEN
7640      ' kari_oki
7650      FOR K9=1 TO GRK(X)
7660        IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9):YO=GI
7670        M(TA,YO)=GRSU(X,K9)
7680      NEXT K9
7690      KTY=TTY
7700      '
7710      TTY=3:GOSUB *MKGRP
7720      ' M_modosi
7730      FOR K9=1 TO GRK(X)
7740        IF KTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9):YO=GI
7750        M(TA,YO)=0
7760      NEXT K9
7770    END IF
7780  NEXT X
7790  ' M_modosi
7800  FOR K9=1 TO GRK(W)
7810    IF KDTY=1 THEN TA=DI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=DI
7820    M(TA,YO)=0
7830  NEXT K9
7840  NEXT W
7850  RETURN
7860 ::
```

```
7900 *MKGRP
7910 ' aki_check
7920 GOSUB *AKICH
7930 IF (AKI=2)OR(AKI=3) THEN
7940 ' -- block^narabi_check
7950 GOSUB *BLNARCH
7960 IF GSY=1 THEN
7970 ' kiroku_zumi?
7980 IF TTY<3 THEN QTY=TTY:QI=I:QJ=HB ELSE QTY=HGR:QI=AA(1,HGR):QJ=FNBL(AA(1,3-HGR))
7990 FOR Z=1 TO GKO
8000 IF (QTY=GRTY(Z))AND(QI=GRI(Z))AND(QJ=GRJ(Z)) THEN RETURN
8010 NEXT Z
8020 ' kiroku
8030 GKO=GKO+1
8040 IF MKGRPN=1 THEN GRD(GKO)=0:GRP(GKO)=2
8050 IF MKGRPN=2 THEN GRD(GKO)=GD+1:GRP(GKO)=GP+1
8060 IF MKGRPN=3 THEN
8070 GRD(GKO)=3
8080 IF DP>GP THEN GRP(GKO)=DP+1 ELSE GRP(GKO)=GP+1
8090 END IF
8100 GRTY(GKO)=QTY:GRI(GKO)=QI:GRJ(GKO)=QJ
8110 GRK(GKO)=AKI
8120 FOR K9=1 TO AKI
8130 IF TTY<3 THEN GRIT(GKO,K9)=A(K9) ELSE GRIT(GKO,K9)=AA(K9,3-HGR)
8140 GRSU(GKO,K9)=GS(K9)
8150 NEXT K9
8160 END IF
8170 END IF
8180 RETURN
8190 :::
8200 *GIHOU4
8210 GIHOU=4
8220 FOR W=1 TO GKO
8230 TTY=GRTY(W):GI=GRI(W):GJ=GRJ(W)
8240 ' kari_oki
8250 FOR K9=1 TO GRK(W)
8260 IF TTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
```

```
8270    M(TA,YO)=GRSU(W,K9)
8280    NEXT K9
8290    KTY=TTY
8300    ' retu
8310    GRKP=GRP(W)
8320    I=GI:GOSUB *TECH3
8330    IF TUIKA=0 THEN
8340        ' block
8350        TTY=3:BI=FNBL(GI):BJ=GJ:GOSUB *TECH3
8360    END IF
8370    ' M_modosi
8380    FOR K9=1 TO GRK(W)
8390        IF KTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
8400        M(TA,YO)=0
8410    NEXT K9
8420    IF TUIKA=1 THEN RETURN
8430    NEXT W
8440    RETURN
8450    :::::
8500 *PAIRLIST
8510    PAIRSU=0
8520    ' -- type_1 --
8530    PRTYPE=1
8540    GOSUB *NTSET
8550    FOR G9=0 TO 26
8560        TTY=INT(NT(1,G9)/10): I=NT(1,G9) MOD 10
8570        IF TTY=3 THEN BI=INT((I-1)/3)+1: BJ=((I-1) MOD 3)+1
8580        GOSUB *SAGASU
8590    NEXT G9
8600    ' -- type_2 --
8610    PRTYPE=2
8620    FOR W=1 TO GKO
8630        TTY=GRTY(W):GI=GRI(W):GJ=GRJ(W)
8640        ' kari_oki
8650        FOR K9=1 TO GRK(W)
8660            IF TTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
8670            M(TA,YO)=GRSU(W,K9)
```



```
8680 NEXT K9
8690 KTY=TTY
8700 ' retu
8710 I=GI:GOSUB *SAGASU
8720 ' block
8730 TTY=3:BI=FNBL(GI):BJ=GJ
8740 GOSUB *SAGASU
8750 ' M_modosi
8760 FOR K9=1 TO GRK(W)
8770 IF KTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
8780 M(TA,YO)=0
8790 NEXT K9
8800 NEXT W
8810 ' -- type_3 --
8820 PRTYPE=3
8830 '
8840 FOR TY=1 TO 2
8850 FOR II=1 TO 3
8860 TE2TY=2: GOSUB *TECH2
8870 NEXT II
8880 NEXT TY
8890 RETURN
8900 ::
8910 *SAGASU
8920 GOSUB *AKICH
8930 IF AKI=2 THEN
8940 ' pair_iti & su
8950 FOR I5=1 TO 2
8960 IF TTY=1 THEN TA=I:YO=A(I5)
8970 IF TTY=2 THEN TA=A(I5):YO=I
8980 IF TTY=3 THEN TA=AA(I5,1):YO=AA(I5,2)
8990 PRI(I5)=9*(TA-1)+YO:PRS(I5)=GS(I5)
9000 NEXT I5
9010 ' kiroku
9020 GOSUB *KIROKU
9030 END IF
9040 RETURN
```

```
9050  ::
9060  *KIROKU
9070  ' kiroku_zumi?
9080  FOR K9=1 TO PAIRSU
9090    IF (PRI(1)=PAIRI(K9,1))AND(PRI(2)=PAIRI(K9,2)) THEN RETURN
9100  NEXT K9
9110  ' kiroku
9120  PAIRSU=PAIRSU+1
9130  PAIRI(PAIRSU,0)=PRTYPE
9140  FOR L9=1 TO 2
9145    PAIRI(PAIRSU,L9)=PRI(L9):PAIRS(PAIRSU,L9)=PRS(L9)
9148  NEXT L9
9150  RETURN
9160  ::
9200 *KOHOSSET
9210  SENUM=SENUM+1
9220  IF SENUM>PAIRSU THEN FKANO=1: RETURN
9230  MGH7SU=MGH7SU+1
9240  IF PAIRI(SENUM,0)>MGH7TY THEN MGH7TY=PAIRI(SENUM,0)
9250  '
9260  TA=INT((PAIRI(SENUM,1)-1)/9)+1:YO=((PAIRI(SENUM,1)-1) MOD 9)+1
9270  SU=PAIRS(SENUM,1): GOSUB *KAKUTEI
9280  RETURN
9290  ::
9300 *KOHOSSEN
9310  ' koho_sentakku
9320  IF KOYO<9 THEN STA=KOTA:SYO=KOYO+1 ELSE STA=KOTA+1:SYO=1
9330  FOR TA=STA TO 9
9340    IF TA>STA THEN SYO=1
9350    FOR YO=SYO TO 9
9360      IF M(TA,YO)=0 THEN
9370        ' koho_sagasi
9380        GOSUB *SIYOUSUU
9390        IF PN=7 THEN KOTA=TA:KOYO=YO: GOTO *EKOHOSAGASI
9400      END IF
9410    NEXT YO
9420  NEXT TA
```

```
9430 IF TA=10 THEN FKANO=1:RETURN
9440 *EKOHOSAGASI
9450 ' kari_oki
9460 MGH7SU=MGH7SU+1
9470 KOSU(1)=TSU(1):KOSU(2)=TSU(2)
9480 TA=KOTA:YO=KOYO:SU=KOSU(1): GOSUB *KAKUTEI
9490 RETURN
9500 :::::
10000 *KIJYUN
10010 ' hyokati_keisan
10020 GOSUB *HYOKATEN
10030 ' level_kettei
10040 SELECT MGIHO
10050 CASE 1
10060     SELECT MAKIN
10070     CASE 1
10080         LEVSU=1
10090     CASE 2
10100         IF MSU(1,2)<=2 THEN LEVSU=1 ELSE LEVSU=2
10110     CASE 3
10120         IF MSU(1,3)<=3 THEN LEVSU=2 ELSE LEVSU=3
10130     CASE 4
10140         IF MSU(1,4)<=3 THEN LEVSU=3 ELSE LEVSU=4
10150     END SELECT
10160 CASE 2
10170     IF HYOKATI<50 THEN LEVSU=4 ELSE LEVSU=5
10180 CASE 3
10190     IF MGH7TY>=2 THEN
10200         SELECT MGH7SU
10210         CASE 1,2
10220             IF HYOKATI<50 THEN LEVSU=5 ELSE LEVSU=6
10230         CASE 3,4,5
10240             LEVSU=6
10250         CASE ELSE
10260             LEVSU=7
10270     END SELECT
10280 ELSE
```

```
10290     IF HYOKATI<50 THEN LEVSU=4
10300     IF (50<=HYOKATI)AND(HYOKATI<100) THEN LEVSU=5
10310     IF 100<=HYOKATI THEN LEVSU=6
10320     END IF
10330     CASE 4
10340     IF FKANO=1 THEN LEVSU=7: GOTO *OUTSEL
10350     SELECT MGH7SU
10360     CASE 0,1,2
10370     IF HYOKATI<100 THEN LEVSU=6 ELSE LEVSU=7
10380     CASE ELSE
10390     LEVSU=7
10400     END SELECT
10410 END SELECT
10420 *OUTSEL
10430 ' level_hyoji
10440 LOCATE 1,22
10450 PRINT "KIJYUN= ";KIJYUN$(LEVSU)
10460 RETURN
10470 :::
11000 *HYOKATEN
11010 ' hyokasu_yomi
11020 FOR I=1 TO 8: READ HYOT(I): NEXT I
11030 ' hyokaten_keisan
11040 HYOTEN=0
11050 FOR I=1 TO 4: HYOTEN=HYOTEN+HYOT(I)*MSU(1,I+2): NEXT I
11060 FOR I=5 TO 8: HYOTEN=HYOTEN+HYOT(I)*MSU(2,I-3): NEXT I
11070 HYOTEN=HYOTEN+4*MGH7SU
11080 'hyokati
11090 HYOKATI=INT(HYOTEN/10)
11100 RETURN
11110 ::
11120 ' --- data ---
11130 DATA 11.5,17.24,32, 8,15.5,21,28
```

(注意 4.1) 技法 7 の選択対として対リストを用いたものを技法 7A と呼び、これを核とする（上のプログラムでは 1760-2000 の部分にあたる）プログラムをプログラム A という。

なお、上記プログラムはまず技法 7A を使って問題を解き（すなわちプログラム A を実行）、解けない時つぎの 5 節で紹介する技法 7B に切り替えてプログラム B を実行する。

5. 超難易問題（レベル6とレベル7）についての考察

[1]の作成段階ではレベル5を超える問題を扱わなかったが、以下では[4]での101題（レベル6が40題、レベル7が61題）について若干の考察をしたい。

まず、4節のコンピュータプログラムAによる基準結果をあげる。

	プログラムAによる基準結果			計
	レベル4	レベル5	解けない	
レベル6	7	16	17	40
レベル7	0	24	37	61

解けない割合の高さ（レベル6は43％，レベル7は61％）から、レベル6とレベル7の難易の妥当性がうなずける（4節で扱ったレベル5の34題の内、解けないのは1題であった）。

（定義）プログラムAで解けないものを、*super\_hard*－タイプとよぶ。

さて、*super\_hard*－タイプのものに基準をつけるには、これを解かなければならない。そのために、空きマス技法で用いる最大空きマス数を5とすると同時に、技法7で用いる候補対の選択に3節で導入したものより広い対象を含む[1]のものをを使う。この選択法による候補対は3節での対リストはもちろんのこと、その他様々なテクニックを用いてやっと見つけれられるものまで含んでいる。

なお、この選択法を技法7Bと呼び、これを核とする（4.2のプログラムで2030-2210の部分にあたる）プログラムをプログラムBという。

（注意5.1）追加されたものについて、1回あたりの使用（手間）ポイントをつぎとする。

- ① 5-空きマス技法：24    ②技法4での4-空きマス技法：21

5.1 上表のプログラムAでレベル5と認定された40題についての基準

- ① 40題とも技法7Aが使用されている。
- ②  $\beta$ -対を使用していないものについての評価値の度数分布表

	50未満	50～100	100以上	計
レベル6	0	1	1	2
レベル7	0	5	1	6

- ③  $\beta$ -対を使用しているものについての技法7Aの使用回数

	1回	2回	3回	4回	5回	6回以上	計
レベル6	5	4	2	0	2	1	14
レベル7	2	5	0	3	1	7	18

④  $\beta$ -対を使用しているものについての評価値の度数分布表

	50 未満	50～100	100 以上	計
レベル 6	7	4	5	14
レベル 7	7	4	7	18

以上の結果を踏まえ、つぎの基準が得られる。

## [基準Ⅳ] プログラム A で解けるものについての基準

$\beta$ -対の使用	技法 7A の使用回数	評価値	基準
no	0 回	50 以上	レベル 5
	1 回以上	50～100 未満	レベル 5
		100 以上	レベル 6
yes	1～2 回	50 未満	レベル 5
		50 以上	レベル 6
	3～5 回	-----	レベル 6
	6 回以上	-----	レベル 7

5.2 *super\_hard* - タイプのものについての基準

まず、プログラム B による結果を示す。

① 解けないものがレベル 6 で 1 題、レベル 7 で 5 題、合計 6 題ある。

② 技法 7B の使用回数 (解けないものは 5 回以上に入れた)

	0 回	1 回	2 回	3 回	4 回	5 回以上	計
レベル 6	2	6	4	0	2	3	17
レベル 7	0	12	8	3	1	13	37

③ 上の技法 7B の使用回数が 0～2 回のもの の評価値の度数分布表

	50 未満	50～100	100 以上	計
レベル 6	1	8	3	12
レベル 7	4	9	7	20

以上の結果を踏まえ、つぎの基準が得られる。

[基準Ⅴ] *super\_hard* - タイプのものについての基準

技法 7B の使用回数	評価値	基準
0～2 回	100 未満	レベル 6
	100 以上	レベル 7
3 回以上	-----	レベル 7
解けない		レベル 7

(注意 5.2) 上で定めた新しい基準による 101 題のレベル付けは次のようになる。

	レベル 4	レベル 5	レベル 6	レベル 7	計
レベル 6	7	7	17	9	40
レベル 7	0	10	20	31	61

## 文 献

- [1] 佐藤金吾, パズル「ナンバープレイス」の難易度について, 法政大学多摩研究報告 16(2001)
- [2] 『ナンプレメイト』(2000 年 10 月号 マガジン・マガジン社)
- [3] 『ナンプレメイト』(2000 年 12 月号 マガジン・マガジン社)
- [4] 『ナンプレ 超上級編 2』(2001 年 10 月 世界文化社)